



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

IN RE APPLICATION OF: Takashi MIYAMORI

GAU:

SERIAL NO: New Application

EXAMINER:

FILED: Herewith

FOR: PROCESSOR CORE FOR USING EXTERNAL EXTENDED ARITHMETIC UNIT EFFICIENTLY AND
PROCESSOR INCORPORATING THE SAME

REQUEST FOR PRIORITY

ASSISTANT COMMISSIONER FOR PATENTS
WASHINGTON, D.C. 20231

SIR:

- ☐ Full benefit of the filing date of U.S. Application Serial Number, filed, is claimed pursuant to the provisions of 35 U.S.C. §120.
- ☐ Full benefit of the filing date of U.S. Provisional Application Serial Number, filed, is claimed pursuant to the provisions of 35 U.S.C. §119(e).
- ☒ Applicants claim any right to priority from any earlier filed applications to which they may be entitled pursuant to the provisions of 35 U.S.C. §119, as noted below.

In the matter of the above-identified application for patent, notice is hereby given that the applicants claim as priority:

<u>COUNTRY</u>	<u>APPLICATION NUMBER</u>	<u>MONTH/DAY/YEAR</u>
Japan	11-269532	September 22, 1999

-----Certified copies of the corresponding Convention Application(s)

- ☒ are submitted herewith
- ☐ will be submitted prior to payment of the Final Fee
- ☐ were filed in prior application Serial No. filed
- ☐ were submitted to the International Bureau in PCT Application Number .
Receipt of the certified copies by the International Bureau in a timely manner under PCT Rule 17.1(a) has been acknowledged as evidenced by the attached PCT/IB/304.
- ☐ (A) Application Serial No.(s) were filed in prior application Serial No. filed ; and
(B) Application Serial No.(s)
 - ☐ are submitted herewith
 - ☐ will be submitted prior to payment of the Final Fee

Respectfully Submitted,

OBLON, SPIVAK, McCLELLAND,
MAIER & NEUSTADT, P.C.

Marvin J. Spivak
Registration No. 24,913

C. Irvin McClelland
Registration Number 21,124



22850

日本国特許庁
PATENT OFFICE
JAPANESE GOVERNMENT

601
JCS905 U.S. PRO
09/666456

別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出願年月日
Date of Application:

1999年 9月22日

出願番号
Application Number:

平成11年特許願第269532号

願人
Applicant(s):

株式会社東芝

2000年 8月 4日

特許庁長官
Commissioner,
Patent Office

及川耕造



CERTIFIED COPY OF
PRIORITY DOCUMENT

出証番号 出証特2000-3062197

【書類名】 特許願

【整理番号】 4HA996099

【提出日】 平成11年 9月22日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/78

【発明の名称】 プロセッサコア及びこれを用いたプロセッサ

【請求項の数】 8

【発明者】

 【住所又は居所】 神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝
 マイクロエレクトロニクスセンター内

 【氏名】 宮森 高

【特許出願人】

 【識別番号】 000003078

 【氏名又は名称】 株式会社 東芝

【代理人】

 【識別番号】 100083806

 【弁理士】

 【氏名又は名称】 三好 秀和

 【電話番号】 03-3504-3075

【選任した代理人】

 【識別番号】 100068342

 【弁理士】

 【氏名又は名称】 三好 保男

【選任した代理人】

 【識別番号】 100100712

 【弁理士】

 【氏名又は名称】 岩△崎▽ 幸邦

【選任した代理人】

 【識別番号】 100100929

【弁理士】

【氏名又は名称】 川又 澄雄

【選任した代理人】

【識別番号】 100108707

【弁理士】

【氏名又は名称】 中村 友之

【選任した代理人】

【識別番号】 100095500

【弁理士】

【氏名又は名称】 伊藤 正和

【選任した代理人】

【識別番号】 100101247

【弁理士】

【氏名又は名称】 高橋 俊一

【選任した代理人】

【識別番号】 100098327

【弁理士】

【氏名又は名称】 高松 俊雄

【手数料の表示】

【予納台帳番号】 001982

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 プロセッサコア及びこれを用いたプロセッサ

【特許請求の範囲】

【請求項 1】 外部に接続される拡張演算ユニットを制御するための命令である拡張演算ユニット制御命令を実装したプロセッサコアであって、

命令をデコードする機能を有し、そのデコードした命令が前記拡張演算ユニット制御命令であった場合には、前記拡張演算ユニットに対して、演算データ、拡張演算コードの動作を規定する命令コード、及び該命令コードが有効であることを示す情報を出力する命令デコード部を備えたことを特徴とするプロセッサコア。

【請求項 2】 前記拡張演算ユニットに対して出力する前記演算データは、該拡張演算ユニットを制御する命令の一部で指定されたレジスタ番号に従って、プロセッサ内部のレジスタファイルから読み出された値であることを特徴とする請求項 1 記載のプロセッサコア。

【請求項 3】 パイプライン停止信号を前記拡張演算ユニットへ出力して、該パイプライン停止信号が有効なときに、前記拡張演算ユニットの実行を一時停止する手段を有することを特徴とする請求項 1 記載のプロセッサコア。

【請求項 4】 パイプラインフラッシュ信号を前記拡張演算ユニットへ出力して、該パイプラインフラッシュ信号が有効なときに、前記拡張演算ユニットへ出力した命令の実行を破棄する手段を有することを特徴とする請求項 1 記載のプロセッサコア。

【請求項 5】 前記拡張演算ユニットからパイプライン停止信号を入力し、該パイプライン停止信号が有効なときに、命令の実行を一時停止する手段を有することを特徴とする請求項 1 記載のプロセッサコア。

【請求項 6】 複数の演算器を有し、前段の演算器の処理結果を次のクロックの立上りで格納する第 1 のパイプラインレジスタと、命令の実行をパイプライン的に処理する第 2 のパイプラインレジスタとを有する拡張演算ユニットと、

命令をデコードする機能を有し、デコードした命令が前記拡張演算ユニットを制御する命令であった場合には、前記拡張演算ユニットに対して、演算データ、

拡張演算コードの動作を規定する命令コード、及び該命令コードが有効であることを示す情報を出力する命令デコード部を有するプロセッサコアとを備えたことを特徴とするプロセッサ。

【請求項 7】 前記拡張演算ユニットから、アドレス、データ、データ書き込みを制御する書き込み制御信号、及びデータ読み出しを制御する読み出し制御信号を入力し、前記読み出し制御信号が有効でデータ読み出しを行う時には、前記アドレスで指定された領域からデータを読み出して前記拡張演算ユニットへ出力し、前記書き込み制御信号が有効でデータ書き込みを行う場合には、前記アドレスで指定された領域に、前記拡張演算ユニットからのデータを書き込むメモリを有することを特徴とする請求項 6 記載のプロセッサ。

【請求項 8】 前記メモリは、前記プロセッサコア及び前記拡張演算ユニットからアクセス可能に構成したことを特徴とする請求項 7 記載のプロセッサ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、命令を実行するプロセッサコア、及びこれを用いたプロセッサに関する。

【0002】

【従来の技術】

従来、この種の分野の技術に関しては、例えば図 7 に示すようなものがあった。

【0003】

図 7 は、従来のプロセッサの基本構成を示すブロック図である。

【0004】

このプロセッサは、プロセッサコア 70 とデータメモリ 80 とを備えている。

【0005】

プロセッサコア 70 は、フェッチ pC (fPe) 701、アドレスインクリメンタ 702、命令メモリ 703、命令レジスタ 704、レジスタファイル 705、デコード回路 706、ソース (1) レジスタ 707、ソース (2) レジスタ 70

8、実行ステージ・コントロールレジスタ710、演算器711、パイプライン制御回路712、メモリステージ・データレジスタ713、メモリステージ・コントロールレジスタ715、レジスタライトステージ・レジスタ716、レジスタライトステージ・コントロールレジスタ717、及び命令有効レジスタ718から構成されている。

【0006】

プロセッサコア70は、5段のパイプライン構成で命令を実行する。すなわち、パイプラインステージ1（命令フェッチステージ）と、パイプラインステージ2（レジスタリードステージ）と、パイプラインステージ3（実行ステージ）と、パイプラインステージ4（メモリステージ）と、パイプラインステージ5（レジスタライトステージ）とを有する。

【0007】

ここで、命令フェッチステージは、命令メモリ703から命令を読み出すステージであり、レジスタリードステージは、命令フェッチステージで読み出した命令で指定された、レジスタファイル705中のレジスタを読み出すとともに、デコード回路706で命令をデコードするステージである。実行ステージは、レジスタリードステージで読み出したレジスタの値と命令のデコード情報によって、演算器711で命令を実行するステージであり、メモリステージは、メモリロード、ストア命令の場合にはデータメモリ80をアクセスする。レジスタライトステージは、実行ステージでの実行結果あるいはメモリロード命令の場合のロードデータをレジスタファイル705へ書き込む。

【0008】

【発明が解決しようとする課題】

しかしながら、従来のプロセッサコアでは、予め定義された命令を実行するだけで、外部に演算ユニットを接続するための拡張機能がなかった。すなわち、外部に演算ユニットを効率よく接続するための機構やインターフェース信号がないため、応用システムに適した演算ユニットを効率よくプロセッサコアに接続して、システム性能の向上を図ることができない。

【0009】

そこで、応用システムに適した演算ユニット、例えば積和演算器をプロセッサコアに予め内蔵することも考えられるが、応用システムによっては、その積和演算器を全く実行しないものもある。従って、プロセッサコアに一樣に前記積和演算器を内蔵することは無駄なハードウェアとなり、コスト増大の要因となる。

【0010】

このようなことから、従来のプロセッサコアでは、コア外部に演算ユニットを効率よく接続するための拡張機能が求められていた。

【0011】

本発明は、上述の如き従来の問題点を解決するためになされたもので、その目的は、外部に演算ユニットを効率よく接続することができるプロセッサコアと、これを用いたプロセッサを提供することである。

【0012】

【課題を解決するための手段】

上記目的を達成するために、請求項1記載の発明に係るプロセッサコアでは、外部に接続される拡張演算ユニットを制御するための命令である拡張演算ユニット制御命令を実装したプロセッサコアであって、命令をデコードする機能を有し、そのデコードした命令が前記拡張演算ユニット制御命令であった場合には、前記拡張演算ユニットに対して、演算データ、拡張演算コードの動作を規定する命令コード、及び該命令コードが有効であることを示す情報を出力する命令デコード部を備えたことを特徴とする。

【0013】

請求項2記載の発明に係るプロセッサコアでは、請求項1記載のプロセッサコアにおいて、前記拡張演算ユニットに対して出力する前記演算データは、該拡張演算ユニットを制御する命令の一部で指定されたレジスタ番号に従って、プロセッサ内部のレジスタファイルから読み出された値であることを特徴とする。

【0014】

請求項3記載の発明に係るプロセッサコアでは、請求項1記載のプロセッサコアにおいて、パイプライン停止信号を前記拡張演算ユニットへ出力して、該パイプライン停止信号が有効なときに、前記拡張演算ユニットの実行を一時停止する

手段を有することを特徴とする。

【0015】

請求項4記載の発明に係るプロセッサコアでは、請求項1記載のプロセッサコアにおいて、パイプラインフラッシュ信号を前記拡張演算ユニットへ出力して、該パイプラインフラッシュ信号が有効なときに、前記拡張演算ユニットへ出力した命令の実行を破棄する手段を有することを特徴とする。

【0016】

請求項5記載の発明に係るプロセッサコアでは、請求項1記載のプロセッサコアにおいて、前記拡張演算ユニットからパイプライン停止信号を入力し、該パイプライン停止信号が有効なときに、命令の実行を一時停止する手段を有することを特徴とする。

【0017】

請求項6記載の発明に係るプロセッサでは、複数の演算器を有し、前段の演算器の処理結果を次のクロックの立上りで格納する第1のパイプラインレジスタと、命令の実行をパイプライン的に処理する第2のパイプラインレジスタとを有する拡張演算ユニットと、命令をデコードする機能を有し、デコードした命令が前記拡張演算ユニットを制御する命令であった場合には、前記拡張演算ユニットに対して、演算データ、拡張演算コードの動作を規定する命令コード、及び該命令コードが有効であることを示す情報を出力する命令デコード部を有するプロセッサコアとを備えたことを特徴とする。

【0018】

請求項7記載の発明に係るプロセッサでは、請求項6記載のプロセッサにおいて、前記拡張演算ユニットから、アドレス、データ、データ書き込みを制御する書き込み制御信号、及びデータ読み出しを制御する読み出し制御信号を入力し、前記読み出し制御信号が有効でデータ読み出しを行う時には、前記アドレスで指定された領域からデータを読み出して前記拡張演算ユニットへ出力し、前記書き込み制御信号が有効でデータ書き込みを行う場合には、前記アドレスで指定された領域に、前記拡張演算ユニットからのデータを書き込むメモリを有することを特徴とする。

【 0 0 1 9 】

請求項 8 記載の発明に係るプロセッサでは、請求項 7 記載のプロセッサにおいて、前記メモリは、前記プロセッサコア及び前記拡張演算ユニットからアクセス可能に構成したことを特徴とする。

【 0 0 2 0 】

【発明の実施の形態】

以下、本発明の実施の形態を図面に基づいて説明する。

【 0 0 2 1 】

[第 1 実施形態]

図 1 は、本発明の第 1 実施形態に係るプロセッサの構成を示すブロック図である。

【 0 0 2 2 】

本実施形態のプロセッサは、プロセッサコア 1 0 と、このプロセッサコア 1 0 の外部に接続された拡張演算ユニット 2 0 と、データメモリ 3 0 と備えている。

【 0 0 2 3 】

初めに、プロセッサコア 1 0 の説明を行う。

【 0 0 2 4 】

このプロセッサコア 1 0 は、フェッチ PC (f P c) 1 0 1、アドレスインクリメンタ 1 0 2、命令メモリ 1 0 3、命令レジスタ 1 0 4、レジスタファイル 1 0 5、デコード回路 1 0 6、ソース (1) レジスタ 1 0 7、ソース (2) レジスタ 1 0 8、実行ステージ・コントロールレジスタ 1 1 0、演算器 1 1 1、パイプライン制御回路 1 1 2、メモリステージ・データレジスタ 1 1 3、メモリステージ・コントロールレジスタ 1 1 5、レジスタライトステージ・データレジスタ 1 1 6、レジスタライトステージ・コントロールレジスタ 1 1 7、及び命令有効レジスタ 1 1 8 から構成されている。

【 0 0 2 5 】

プロセッサコア 1 0 は、次の 5 段のパイプライン構成で命令を実行する。すなわち、パイプラインステージとして、命令メモリ 1 0 3 から命令を読み出すステージであるパイプラインステージ 1 (命令フェッチステージ) と、命令フェッチ

ステージで読み出した命令で指定された、レジスタファイル 105 中のレジスタを読み出すとともに、デコード回路 106 で命令をデコードするパイプラインステージ 2 (レジスタリードステージ) と、レジスタリードステージで読み出したレジスタの値と命令のデコード情報とによって、演算器 111 で命令を実行するパイプラインステージ 3 (実行ステージ) と、メモリロードやストア命令の場合にデータメモリ 30 をアクセスするパイプラインステージ 4 (メモリステージ) と、実行ステージでの実行結果あるいはメモリロード命令の場合のロードデータをレジスタファイル 105 へ書き込むパイプラインステージ 5 (レジスタライトステージ) とを有する。

【0026】

本実施形態のプロセッサコア 10 の代表的な命令としては、例えば、加算命令 (a d d)、データメモリロード命令 (l o a d)、データメモリストア命令 (s t o r e)、ジャンプ命令 (j u m p)、及び拡張命令 (e x t) があり、それぞれの命令は 32 ビット長とする。

【0027】

図 2 は、これら各命令のビットパターンを示す図であり、同図を参照して各命令の動作を説明する。

【0028】

加算命令 (a d d) は、32 ビット命令のビット 19～16 が 0000 であり、s r c 1 フィールドで指定されるレジスタと s r c 2 フィールドで指定されるレジスタの値とを加算して、d s t フィールドで指定されるレジスタへ書き込む。

【0029】

データメモリロード命令 (l o a d) は、命令のビット 19～16 が 0001 であり、命令の動作は、s r c 1 フィールドで指定されるレジスタの値をデータメモリ 30 のアドレスとして、データメモリ 30 のデータを読み出し、d s t フィールドで指定されるレジスタへ書き込む。

【0030】

データメモリストア命令 (s t o r e) は、命令のビット 19～16 が 001

0であり、命令の動作は、src1フィールドで指定されるレジスタの値をデータメモリ30のアドレスとして、src2フィールドで指定されるレジスタの値をデータメモリ30へ書き込む。

【0031】

ジャンプ命令(jump)は、命令のビット19～16が0011であり、src1フィールドで指定されるレジスタの値を読み出し、ジャンプ先アドレスとしてジャンプを行う。

【0032】

拡張命令(ext)は、命令のビット19～16が0100であり、src1フィールドで指定されるレジスタとsrc2フィールドで指定されるレジスタの値とを読み出し、外部の拡張演算ユニット20に出力する。拡張演算ユニット20では演算結果をプロセッサコア10へ出力し、その結果をdstフィールドで指定されるレジスタへ書き込む。

【0033】

上述した各パイプラインステージの基本的な動作を説明する。

【0034】

＜パイプラインステージ1の動作＞

パイプラインステージ1(命令フェッチステージ)では、フェッチpc101の値により命令メモリ103から命令を読み出す。次のクロックの立上りで、命令メモリ103から読み出された命令が命令レジスタ104へ書き込まれるとともに、命令有効レジスタ118の値は1にセットされる。また、フェッチPC101の値がアドレスインクリメンタ102で命令ワードのバイト数(命令ワードが4バイトの場合は4)だけ加えられ、次のクロックの立上りでフェッチpc101に格納され、次の命令が命令メモリ103から読み出されるという動作を繰り返す。

【0035】

＜パイプラインステージ2の動作＞

パイプラインステージ2(レジスタリードステージ)では、命令レジスタ104の命令で指定されたレジスタの値をレジスタファイル105から読み出す。図

2に示した命令の場合では、命令のビット27～24がsrc1フィールド、ビット23～20がsrc2フィールドであり、このビットで指定されたレジスタの値がレジスタファイル105に読み出される。レジスタファイル105から読み出されたレジスタの値は、次のクロックの立上リエッジで、ソース(1)レジスタ107及びソース(2)レジスタ108に格納される。

【0036】

また、デコード回路106は、命令有効レジスタ118の値が1で命令レジスタ104に有効な命令が格納されている場合に、命令レジスタ104の命令をデコードして実行ステージ・コントロールレジスタ110へ、演算器111の制御情報と、デスティネーションレジスタ番号及びその有効情報と、ジャンプ命令によるパイプラインフラッシュ情報とを出力する。これらの情報は、次のクロックの立上りで、実行ステージ・コントロールレジスタ110に格納される。

【0037】

さらに、デコード回路106の動作を詳細に説明すると、デコード回路106は、命令レジスタ104から32ビットの命令を読み出し、図3に示すように制御情報を出力する。デスティネーションレジスタ番号(dst番号)としては命令のビット31～28を出力する。デスティネーション有効情報は、命令実行の結果、レジスタへの書き込みを行うadd, load, ext命令の場合には1(有効)となる。パイプラインフラッシュ情報(Flush)はジャンプ命令のとき1(有効)となる。拡張命令有効情報(ext有効)はext命令の場合に1(有効)となる。また、拡張命令情報(code16)としては、命令コードのビット15～0が出力される。

【0038】

<パイプラインステージ3の動作>

パイプラインステージ3(実行ステージ)では、演算器111はソース(1)レジスタ107及びソース(2)レジスタ108からデータを入力して、実行ステージ・コントロールレジスタ110内の、演算器111の制御情報に従って演算を行う。演算器111の制御情報は、図3に示したデコード回路106の出力に対応しており、演算器111は図4に示すような動作を行う。

【0039】

演算器 111 の演算結果は、メモリステージ・データレジスタ 113 に格納されるとともに、データメモリ 30 をアクセスするデータロード命令とデータストア命令の場合では演算器 111 の出力をデータメモリ 30 のアドレスとして出力する。データストア命令の場合は、セクタ 109 からの値をデータメモリ 30 へストアするデータとして出力する。

【0040】

実行ステージ・コントロールレジスタ 110 のデスティネーションレジスタ番号とその有効情報が、次のクロックの立上りでメモリステージ・コントロールレジスタ 115 へ格納される。

【0041】

実行ステージ・コントロールレジスタ 110 にジャンプ命令によるパイプラインフラッシュ情報が有効になった場合は、パイプライン制御回路 112 ではパイプラインフラッシュ信号を有効にして、次のクロックの立上りで、命令有効レジスタ 118 及び実行ステージ・コントロールレジスタ 110 の有効情報を無効にして、次のクロックでのデコードステージ及び実行ステージが動作しないように制御する。

【0042】

また、パイプラインフラッシュ信号がアクティブになると、演算器 111 の出力のジャンプ先アドレスが次のフェッチ pc 101 の値になるように制御する。

【0043】

同時に、パイプラインフラッシュ信号を拡張演算ユニット 20 へ出力し、拡張演算ユニット 20 内部の実行ステージレジスタ 201 を無効にできるようにする。

【0044】

パイプライン制御回路 112 は、データメモリ 30 からのパイプラインストール（停止）要求、あるいは拡張演算ユニット 20 からのパイプライン停止要求が発生した場合に、フェッチ pc 101、命令レジスタ 104、命令有効レジスタ 118、ソース（1）レジスタ 107、ソース（2）レジスタ 108、及び実行ステージ・コントロールレジスタ 110 の値が、次のクロックの立上りで更新さ

れないように制御信号を各モジュールへ出力する。

【0045】

データメモリ30からのパイプラインストール要求の場合は、メモリステージ・コントロールレジスタ115の値も更新されないように、メモリステージ・コントロールレジスタ115へ制御信号を出力する。同時に、パイプライン制御回路112は、拡張演算ユニット20へパイプライン停止信号を出力し、拡張演算ユニット20内部の実行ステージレジスタ201の値が更新されないように制御する。

【0046】

〈パイプラインステージ4の動作〉

パイプラインステージ4（メモリステージ）では、データロード命令の場合はデータメモリ30からデータが読み出され、次のクロックでレジスタライトステージ・データレジスタ116に格納される。その他の命令の場合は、メモリステージ・データレジスタ113の値が、次のクロックでレジスタライトステージ・データレジスタ116に格納される。また、メモリステージ・コントロールレジスタ115の情報が次のクロックで、レジスタライトステージ・コントロールレジスタ117に格納される。

【0047】

〈パイプラインステージ5の動作〉

パイプラインステージ5（レジスタライトステージ）では、レジスタライトステージ・コントロールレジスタ117のデスティネーションレジスタへの書き込みが有効の場合は、レジスタライトステージ・コントロールレジスタ117内のデスティネーションレジスタ番号で指定されるレジスタファイル105のレジスタへ、レジスタライトステージ・データレジスタ116の値を書き込むように制御する。

【0048】

次に、本実施形態に係る拡張演算ユニット20について説明する。

【0049】

拡張演算ユニット20は、プロセッサコア10と接続されており、プロセッサ

コア10のデコードステージで、コア10から、レジスタファイル105より読み出した命令の `src1`、`src2` フィールドに対応するレジスタの値を入力する。

【0050】

デコード回路106から、命令の下位16ビットに相当する拡張命令情報 (`code16`) を入力すると共に、デコードステージの命令が拡張命令 `ext` の場合に有効 (1) になる拡張命令有効情報を入力する。これらの情報は、コア10からのパイプラインフラッシュ信号、あるいはパイプラインストール信号が有効にならない場合は、次のクロックの立上りで、実行ステージレジスタ201へ格納される。プロセッサコア10からのパイプラインストール信号が有効なときは、実行ステージレジスタ201の値を更新しない。

【0051】

プロセッサコア10からのパイプラインフラッシュ信号が有効なときは、次のクロックの立上りで、拡張命令有効情報を強制的に無効 (0) にして、拡張演算ユニット20が演算を行わないように制御する。

【0052】

ステージ (1) 演算器202は、プロセッサコア10のメモリステージに対応したステージで演算を行うユニットである。このステージ (1) 演算器202の動作は、コア10から入力した拡張命令情報 `code16` をコントロール回路204がデコードして行う。またステージ (1) 演算器202は、拡張命令情報の一部を演算の入力データとして使用することもできる。

【0053】

ステージ (1) の演算結果は、次のクロックの立上りでパイプライン (1) レジスタ205に格納される。パイプライン (1) レジスタ205の結果は、次のステージ (2) 演算器203の入力データとして使うことができる。ステージ (2) 演算器204の結果は、次のクロックの立上りで、パイプライン (2) レジスタ206に格納される。

【0054】

本実施形態での拡張命令の例を以下に挙げる。

【 0 0 5 5 】

```

1 ext dst,src1,src2,0x0001 acc <- rc1*src2;dst <- src1
2 ext dst,src1,src2,0x0002 acc <- acc+src1*src2;dst <- src1
3 ext dst,src1,src2,0x0003 dst <- acc

```

これらの命令を実行するため、拡張演算ユニット 2 0 のステージ (1) 演算器 2 0 2 は乗算を行い、ステージ (2) 演算器 2 0 3 は、パイプライン (1) レジスタ 2 0 5 から入力したステージ (1) 演算器 2 0 2 の乗算の結果とパイプライン (2) レジスタ 2 0 6 とを加算する。ステージ (2) 演算器 2 0 3 の結果は、パイプライン (2) レジスタ 2 0 6 (この場合は a c c レジスタ) に再度格納される。

【 0 0 5 6 】

上記 1 の命令では、プロセッサコア 1 0 から入力したソース (1) データとソース (2) データを入力として、ステージ (1) 演算器 2 0 2 で乗算を行う。結果は、次のクロックの立上りでパイプライン (1) レジスタ 2 0 5 に格納される。ステージ (2) 演算器 2 0 3 では、処理を行わず、パイプライン (1) レジスタ 2 0 5 に格納された乗算結果を次のクロックの立上りでパイプライン (2) レジスタ 2 0 6 に格納する。また、ステージ (1) 演算器 2 0 2 から、ソース (1) データを命令の結果としてプロセッサコア 1 0 へ出力する。これは、プロセッサコア 1 0 の実行ステージに相当し、プロセッサコア 1 0 は、上記 1 の命令の結果として、レジスタライトステージでレジスタファイル 1 0 5 へ書き込む。

【 0 0 5 7 】

上記 2 の命令では、ステージ 1 までは、上記 1 の命令と同様に動作する。ステージ (2) 演算器 2 0 3 では、パイプライン (1) レジスタ 2 0 5 の結果とパイプライン (2) レジスタ 2 0 6 の結果とを加算し、その結果をパイプライン (2) レジスタ 2 0 6 へ格納する。また、ステージ (1) 演算器 2 0 2 から、ソース (1) データを命令の結果としてプロセッサコア 1 0 へ出力する。これは、プロセッサコア 1 0 の実行ステージに相当し、プロセッサコア 1 0 は、上記 1 の命令の結果として、レジスタライトステージでレジスタファイル 1 0 5 へ書き込む。

【0058】

上記3の命令では、パイプライン(2)レジスタ206を読み出し、ステージ(1)演算器202から、ソース(1)データを命令の結果としてプロセッサコア10へ出力する。これは、プロセッサコア10の実行ステージに相当し、プロセッサコア10は、上記1の命令の結果として、レジスタライトステージでレジスタファイル105へ書き込む。

【0059】

パイプライン(2)レジスタ206には、上記1あるいは2の命令の結果が格納される。したがって、上記3の命令が実行ステージ・レジスタ201に格納されたとき、パイプライン(1)レジスタ205に上記1あるいは2の命令が格納されることを、コントロール回路204はチェックし、この場合は、1サイクルだけコア10へのパイプラインストール信号を有効にして、パイプラインを停止させ、次のクロックサイクルで、上記1あるいは2の命令の結果を読み出して、ステージ(1)演算器202を介して、コア10へ結果を出力することができる。

【0060】

上述したように、本実施形態では、プロセッサコア10に、拡張演算ユニット20を制御するための拡張命令を備え、この拡張命令を実行するための制御回路と、拡張演算ユニット20を制御しデータを転送するインターフェース回路とを内蔵したので、プロセッサコア10の外部に、効率よく拡張演算ユニット20を接続することができる。

【0061】

例えば、拡張演算ユニット20として、レイテンシが2サイクルでスループットが1サイクルの積和演算器を設けた場合を考える。プロセッサコア10の乗算命令では実行に5サイクル、加算命令は1サイクルかかったとする。10回の積和演算は、拡張演算ユニット20の積和演算器を用いることにより、11サイクルで実行することができる。これに対して、プロセッサコア10の乗算命令と加算命令の組合せによる場合では、 $(5+1) \times 10 = 60$ で、60サイクルかかることになる。

【 0 0 6 2 】

プロセッサコアに予め積和演算器を内蔵することも考えられるが、ある応用システムで積和演算が全く実行されない場合、この応用システムでは積和演算ユニットが無駄なハードウェアとなり、コスト増大の要因となる。

【 0 0 6 3 】

本実施形態では、積和演算の例以外にも、応用システムに適した演算ユニットをプロセッサコア 1 0 の外部に接続することによって、システム性能を向上させることが可能となる。

【 0 0 6 4 】

なお、本実施形態では、拡張演算ユニット 2 0 はステージ (1) 演算器 2 0 2 とステージ (2) 演算器 2 0 3 という 2 段のパイプライン構成を採ったが、拡張演算ユニット 2 0 の構成はこれに限定されるものではなく、3 段以上の多段のパイプラインステージを持つことができる。

【 0 0 6 5 】

また、プロセッサコア 1 0 へのパイプラインストール要求の例として、実行に複数ステージが必要な命令の結果を読み出す場合を示したが、これに限定されるものではない。例えば、内部がパイプライン処理されず、命令の実行に複数サイクルが必要な場合は、次の命令の実行を、現在実行している命令が終了するまで待たせることが考えられる。これは、コントロール回路 2 0 4 がこのような命令実行のステートを管理することで実現できる。

【 0 0 6 6 】

[第 2 実施形態]

図 5 は、本発明の第 2 実施形態に係るプロセッサの構成を示すブロック図であり、図 1 と共通の要素には、同一の符号を付しその説明を省略する。

【 0 0 6 7 】

第 2 実施形態のプロセッサでは、プロセッサコア 1 0 の構成は図 1 と同じであるが、拡張演算ユニット 4 0 のステージ (1) 演算器 4 0 2 からデータメモリ 3 1 へ、アドレス信号、リード／ライトのコントロール信号、及びライトデータ信号が出力され、データメモリ 3 1 を直接アクセスできるようになっている。

【0068】

データメモリ31は、セクタ31a、31bによって、コア10からのアドレス信号、リード／ライトコントロール信号、及びライトデータ信号を選択して、RAMモジュールへ出力することで、コア10と拡張演算ユニット40からのアクセス要求を調停する。

【0069】

データメモリ31から読み出されたデータは、拡張演算ユニット40へ出力され、ステージ(2)演算器403の出力と選択されてパイプライン(2)レジスタ406へ、次のクロックの立上りで格納される。

【0070】

本実施形態の拡張演算ユニット40では、ステージ(3)演算器407が、パイプライン(2)レジスタ406の出力を入力データとして演算に使用している。このように、データメモリ31から読み出した結果を拡張演算ユニット40での演算入力データとして使用することができる。

【0071】

[第3実施形態]

図6は、本発明の第3実施形態に係るプロセッサの構成を示すブロック図であり、図1と共通の要素には、同一の符号を付しその説明を省略する。

【0072】

本実施形態のプロセッサは、図1に示した構成において、拡張演算ユニット20からデスティネーションレジスタ有効ビットを出力することにより、プロセッサコア10の実行ステージでの演算結果をプロセッサコア10で無効にし、レジスタライトステージでレジスタへ書き込まないようにしたものである。

【0073】

本実施形態では、上記第1実施形態で示した拡張命令の例を次のように変更する。

【0074】

```
1 ext src1,src2,0x0001 acc ← src1*src2;
2 ext src1,src2,0x0002 acc ← acc+src1*src2;
```

3 ext dst,src1,src2,0x0003 dst ← acc

上記 1 及び 2 の命令では、プロセッサコア 1 0 のレジスタファイル 1 0 5 へ結果を書き戻さないため、拡張演算ユニット 2 0 のコントロール回路 2 0 4 a は、デスティネーション有効情報を無効として、プロセッサコア 1 0 へ出力する。プロセッサコア 1 0 の実行ステージ・コントロールレジスタ 1 1 0 のデスティネーションレジスタの有効情報は、この場合には無効 (0) として次のクロックの立ち上がりでメモリステージ・コントロールレジスタ 1 1 5 a へ格納される。

【 0 0 7 5 】

上記 3 の命令では、プロセッサコア 1 0 のレジスタファイル 1 0 5 へ結果を書き戻す。このため、拡張演算ユニット 2 0 のコントロール回路 2 0 4 a は、デスティネーション有効情報を有効として、プロセッサコア 1 0 へ出力する。

【 0 0 7 6 】

プロセッサコア 1 0 は、拡張命令を実行している場合に、拡張演算ユニット 2 0 のコントロール回路 2 0 4 a から出力されるデスティネーション有効情報を、次のクロックの立ち上がりでメモリステージ・コントロールレジスタ 1 1 5 a に格納するように制御する。

【 0 0 7 7 】

プロセッサコア 1 0 のパイプライン制御によっては、実行ステージとメモリステージのデスティネーションレジスタが有効で、そのレジスタ番号がレジスタリードステージで読み出すレジスタ番号と一致したとき、パイプラインをストールさせる場合がある。拡張命令で、プロセッサコア 1 0 のレジスタファイル 1 0 5 への書き込みを行わない命令がある場合には、デスティネーションレジスタへの書き戻しを無効という情報を、拡張演算ユニット 2 0 からプロセッサコア 1 0 へ出力する機能を設けることによって、本来不要であるパイプラインのストールを回避することができる。

【 0 0 7 8 】

【発明の効果】

以上詳細に説明したように、本発明よれば、拡張演算ユニットを制御するための命令である拡張演算ユニット命令をサポートし、この拡張演算ユニット命令の

実行を制御する機能と、拡張演算ユニットを接続するインターフェース機能とを備えたので、プロセッサコアの外部に、効率よく拡張演算ユニットを接続することが可能になる。これにより、応用システムに適した拡張演算ユニットを、プロセッサコアの外部に接続することによって、システム性能を容易に向上させることが可能となる。

【図面の簡単な説明】

【図 1】

本発明の第 1 実施形態に係るプロセッサの構成を示すブロック図である。

【図 2】

プロセッサコアの命令のビットパターンを示す図である。

【図 3】

図 1 に示したデコード回路 1 0 6 の動作を示す図である。

【図 4】

図 1 に示した演算器 1 1 1 の動作を示す図である。

【図 5】

本発明の第 2 実施形態に係るプロセッサの構成を示すブロック図である。

【図 6】

本発明の第 3 実施形態に係るプロセッサの構成を示すブロック図である。

【図 7】

従来のプロセッサの基本構成を示すブロック図である。

【符号の説明】

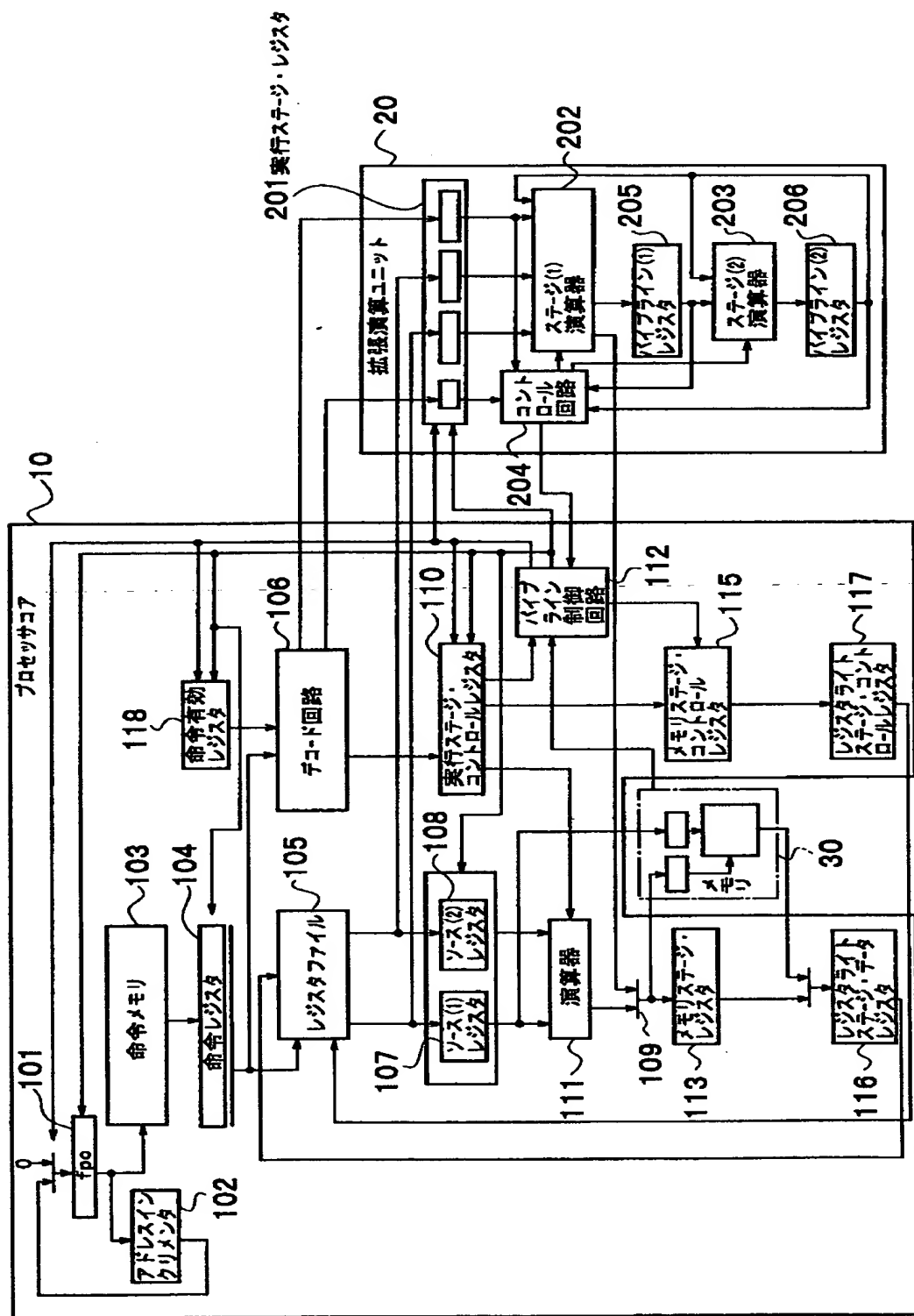
- 1 0 プロセッサコア
- 2 0, 4 0 拡張演算ユニット
- 3 0, 3 1 データメモリ
- 1 0 1 フェッチ P C (f P c)
- 1 0 2 アドレスインクリメンタ
- 1 0 3 命令メモリ
- 1 0 4 命令レジスタ
- 1 0 5 レジスタファイル

- 106 デコード回路
- 107 ソース (1) レジスタ
- 108 ソース (2) レジスタ
- 110 実行ステージ・コントロールレジスタ
- 111 演算器
- 112 パイプライン制御回路
- 113 メモリステージ・データレジスタ
- 115 メモリステージ・コントロールレジスタ
- 116 レジスタライトステージ・データレジスタ
- 117 レジスタライトステージ・コントロールレジスタ
- 118 命令有効レジスタ
- 202, 402 ステージ (1) 演算器
- 203, 403 ステージ (2) 演算器
- 205, 405 パイプライン (1) レジスタ
- 206, 406 パイプライン (2) レジスタ
- 407 ステージ (3) 演算器
- 408 パイプライン (3) レジスタ

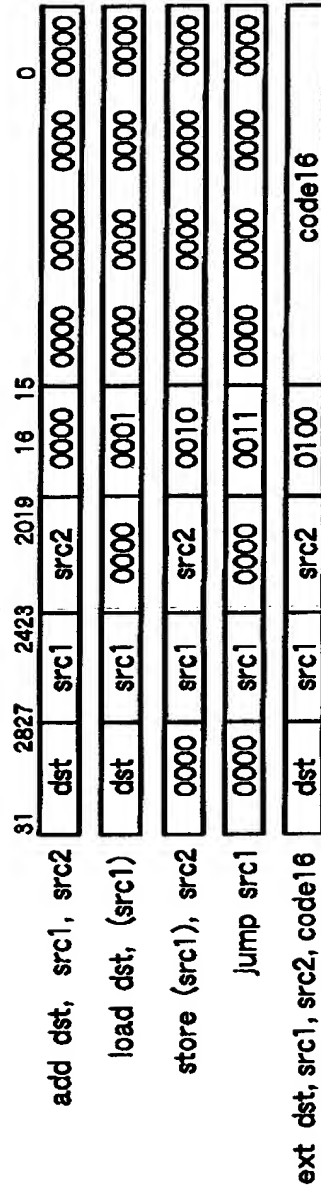
【書類名】

図面

【图 1】



【図 2】



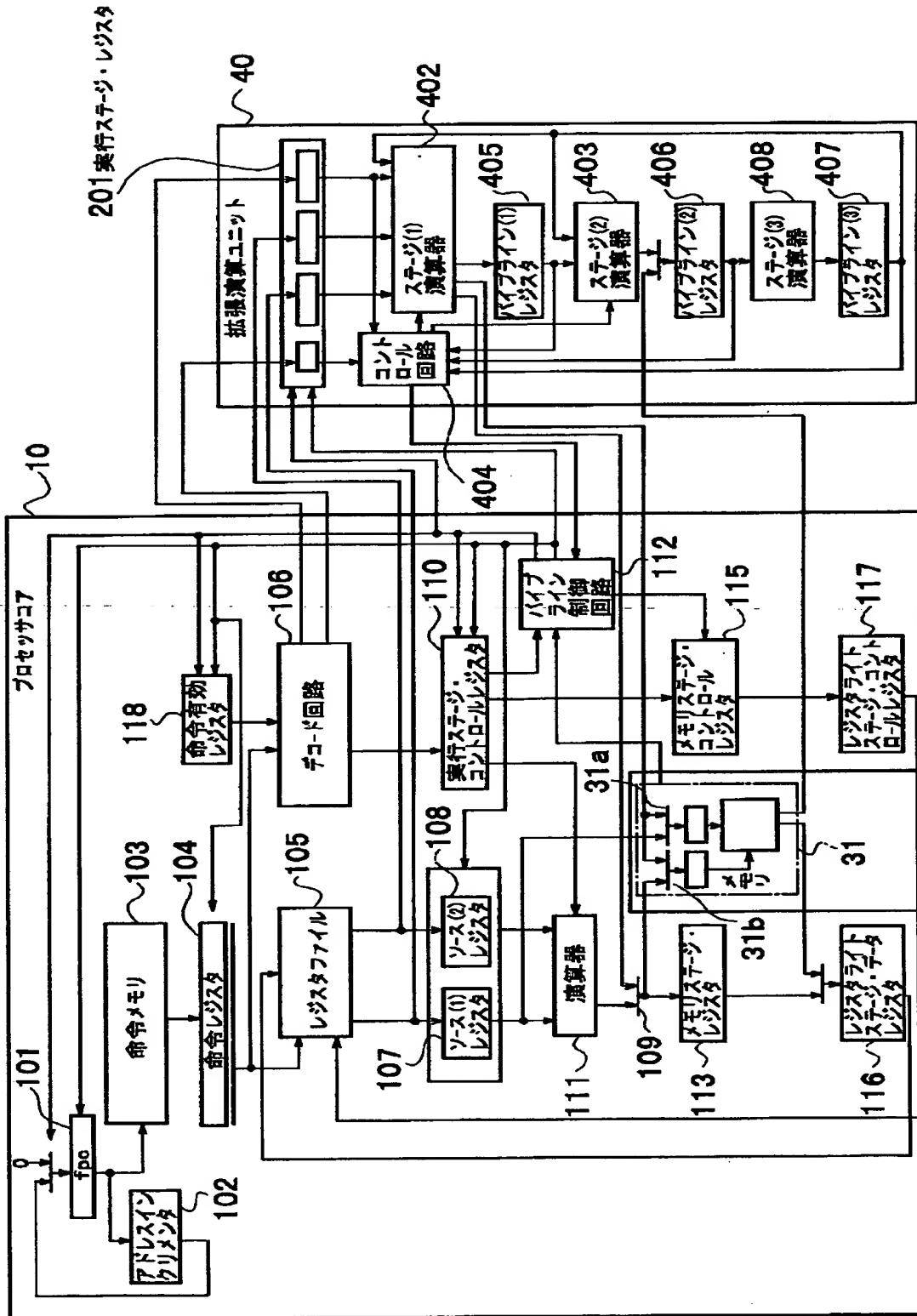
【図 3】

命令入力		出力					
bit	19~16	演算器制御	dst番号	dst有効	Flush	ext有効	code16
add	0000	01(add)	bit31~28	1	0	0	bit15~0
load	0001	00(nop)	bit31~28	1	0	0	bit15~0
store	0010	00(nop)	bit31~28	0	0	0	bit15~0
jump	0011	00(nop)	bit31~28	0	1	0	bit15~0
ext	0100	10(ext)	bit31~28	1	0	1	bit15~0

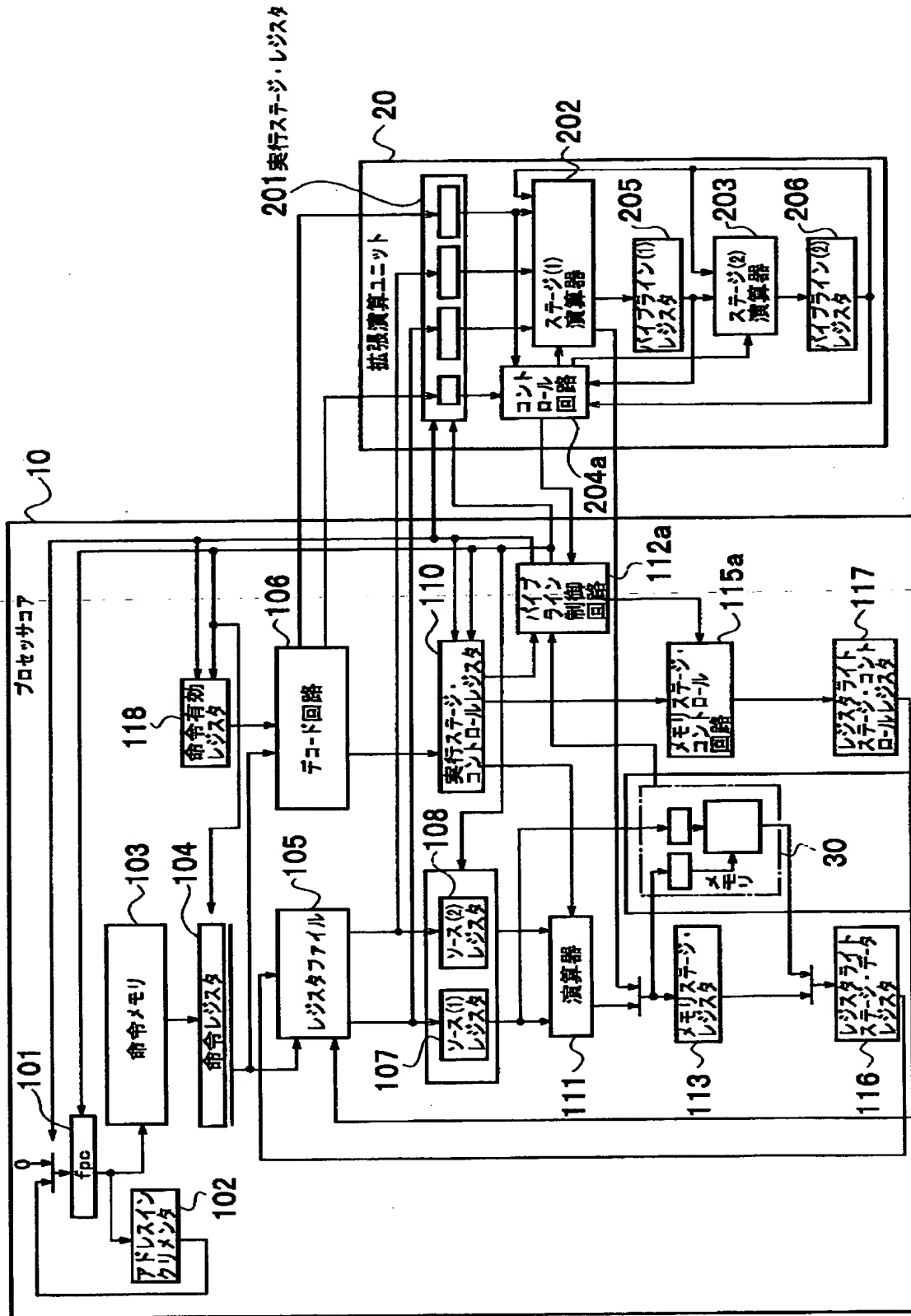
【図 4】

演算器制御入力	演算器出力
00 (nop)	ソ-ス1レジスタ107
01 (add)	ソ-ス1レジスタ107+ソ-ス2レジスタ108
10 (ext)	拡張演算ユニットからの入力

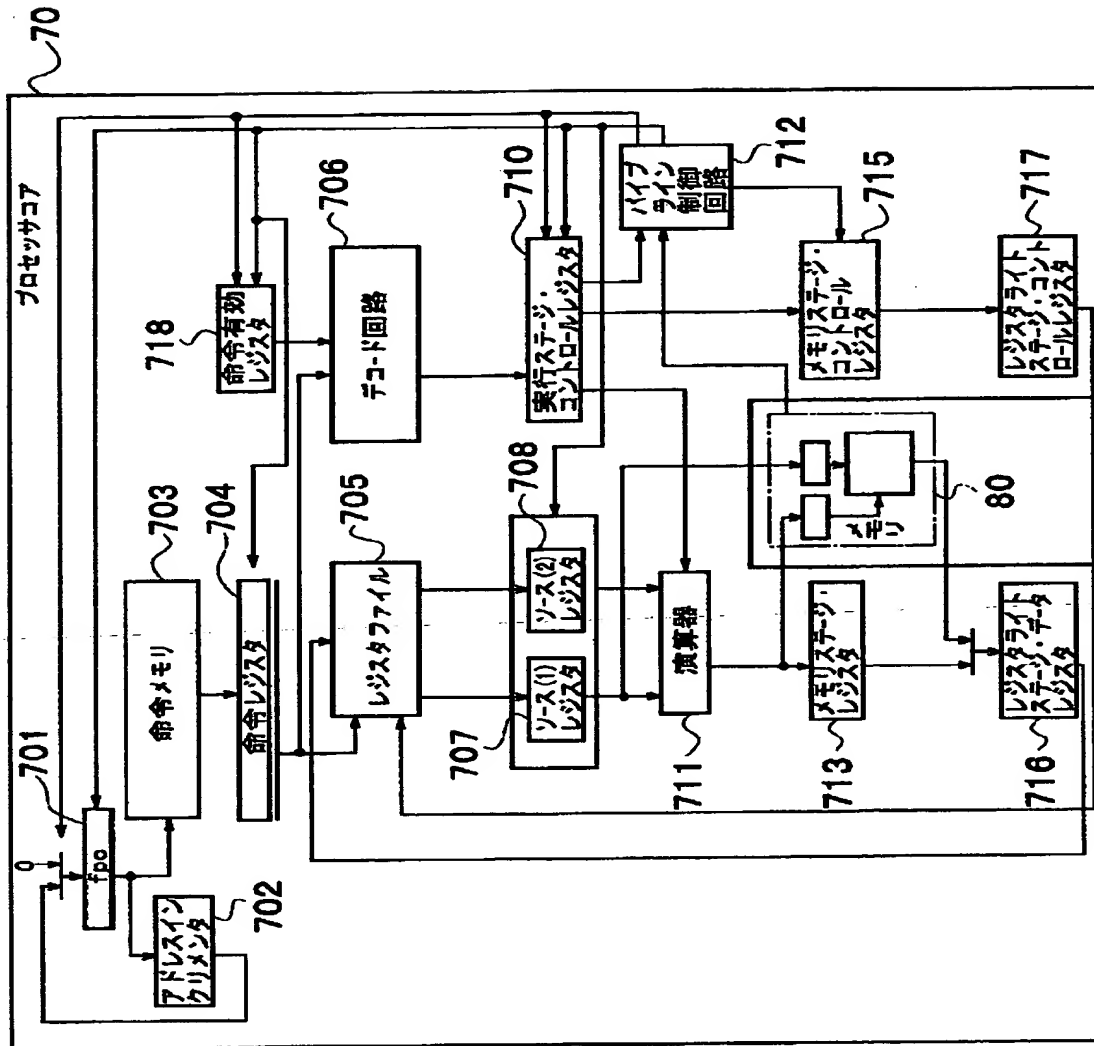
【図 5】



【図 6】



【図 7】



【書類名】 要約書

【要約】

【課題】 外部に演算ユニットを効率よく接続することができるプロセッサコアを提供する。

【解決手段】 外部に接続された拡張演算ユニットを制御するための命令である拡張演算ユニット制御命令を実装したプロセッサコアであって、命令をデコードする機能を有し、そのデコードした命令が前記拡張演算ユニット制御命令であった場合には、前記拡張演算ユニットに対して、演算データ、拡張演算コードの動作を規定する命令コード、及び該命令コードが有効であることを示す情報を出力する命令デコード部を備える。すなわち、プロセッサコアは、拡張演算ユニットを制御するための拡張命令を備え、さらに拡張命令を実行するための制御手段と、拡張演算ユニットを制御しデータを転送するインターフェース手段とを内蔵した。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000003078]

1. 変更年月日	1990年 8月22日
[変更理由]	新規登録
住 所	神奈川県川崎市幸区堀川町72番地
氏 名	株式会社東芝